Universiteit Leiden

ICT in Business and the Public Sector

Effect of negative emotional display on contribution acceptance in open source projects: evidence from GitHub

Name: Ivo Valchev
Student-no: S2435209

Date: 08/08/2021

1st supervisor: Amirhossein Zohrehvand
2nd supervisor: Bas Kruiswijk

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

This paper studies whether communicating negative emotion impacts voluntary contributions to open source. It is the first study within emotional display in open source contributions to research negatively charged communication. It builds upon prior research of positive emotional display to further our understanding of the relationship between emotion and the acceptance rate and processing time of contributions to open source. I construct a dataset of the communication in the largest open source communities to gather empirical evidence. The findings show no evidence of a causation relationship between negative emotion and the outcome state of change requests, but suggest that negative displays increase the time to reach such outcome. The paper outlines information for future studies in that field that may for the first time analyse collaboration across multiple levels of communication. It completes the pathway identified in prior research for the development of a mixed emotional display effect model of open source software development.

Personal note

This paper is the result of a master thesis conducted at the Leiden Institute of Advanced Computer Science, the Netherlands, between 2020 and 2021. During the height of the COVID-19 pandemic, I received the utmost support from the university staff who supported my academic growth and development. For that, I thank in particular my supervisor Assistant Professor Dr. Amirhossein H. Zohrehvand for his constant support, guidance and professionalism. In addition, I thank the other staff members who assisted me in various capacities during this time.

Introduction

Overview

Open-source projects play an important role in the development of software worldwide. Open source projects are often developed and maintained by freelancers and part-time developers. Voluntary contributions are a key factor for the longevity of open source software (OSS) (Roberts et al., 2006). Past research shows emotions in such contributions are important in respect to OSS development, but we don't yet fully understand the effect of emotional display on contribution acceptance. While previous studies have looked into positive emotional display, there remain unaddressed issues with the role of negative emotional display in voluntary open source contributions. This prevents the emergence of a clear picture regarding the roles and effects of emotional display in communities, upon which many major software products and services depend.

This research will help to understand negative emotions by empirical analysis of the communications between various actors of open source collaboration.

Problem statement

How does negative emotional display affect collaborative work geographically dispersed individuals? Does the presence or lack thereof of negative emotions change contribution acceptance in a statistically significant way?

Community-based development is increasingly becoming an alternative, or addition to, the firm-based model (Grewal et al., 2006). However, while the firm-based model is characterised by a structure of hierarchy and authority, other collaboration models lack those qualities.

Voluntary collaborative methods, such as those prevalent in the nature of OSS, show distinct disparities with the traditional firm-based model (Grewal et al., 2006).

The OSS model cannot use authority to achieve results as firms have done in the traditional sense, but rather rely on collaborations and interactions between individual contributors. Previous research notes network embeddedness within an OSS has a powerful effect on the success and longevity of open source projects (Grewal et al., 2006).

While those effects have been studied in the past using a black-box model of inputs and outputs (benchmarks for success), few researchers have studied concrete variables affecting the dynamics of OSS collaboration and communication. Specifically, the role of emotionally-powered written communication, which is characteristic for virtual OSS projects (Werder, 2018), has received limited attention.

The nature of negative emotional display, and its impact, has had an increasingly prominent place with recent technological developments. Social media platforms facilitate the formation and dispersal of negative opinions to millions of people within hours, culminating in *online firestorms*: events that are characterised by sudden and large discharges of messages containing negative statements and emotional displays. How are OSS contributions different?

Existing research (Werder, 2018) suggests that positive emotional displays have shown influence on team processes and performance. However, as of February 2021, as pointed out by the same authors, there is no empirical research that looks into the role of negative emotional displays and its impact on OSS projects over time.

Research question

Building upon the suggested research of Werder, this research paper bases itself upon the following question:

> *What is the effect of merging Pull Requests on GitHub with negative display of emotion on the contribution acceptance rate of OSS?*

Literature Review

The question about the effect of emotion, whether it be positive, negative or other, in collaboration on OSS projects falls within a broader area of extensive study of emotional information in software development. In formal organisations, collaboration is underpinned by the explicit structure and hierarchy of authority that establishes organisational norms for collaboration and communication. OSS projects are distinct in that they often lack explicit hierarchical structures given the voluntary nature of contributions.

Emotional displays are, generally, characterised and studied as positive or negative Werder, K. (2018). Prior studies have researched the impact of emotion along this paradigm, beginning with assumptions that positive emotion is associated with a constructive or beneficial impact, while negative emotion leads to negative association. The study of emotional display through textual analysis is referred to as sentiment analysis, while its output studies refer to as sentiment. The concept of sentiment is widely used in the study of emotionally charged exchanges between members of a community.

An important milestone in the study of emotional display in OSS projects was the finding that issue reports convey emotionally charged information. Furthermore, those involved in the issue creation process (the reporters, commenters and maintainers) recognise that issues contain such emotionally charged information (Murgia et al., 2018).

Moreover, the effects of such expressions of emotion are known in other areas among the web development community. Bazelli et at. (2013) investigated emotional displays of users on OSS platforms which contain information on common programming pitfalls in question-answer format by community members. In their research, they found that top reputed authors who showed negative emotional expressions the least often among the community, tended to get less downvotes than those authors who exhibited such behaviour.

The interactions between a contributor and a maintainer have been studied through the prism of the change requester - maintainer relationship from a variety of angles. The geographic location of the contributor, his or her gender, community status and prior experience (Furtado et al., 2021) all have an impact on the likelihood that a change request gets accepted, and additionally an impact on the longevity of OSS projects over time.

Differences in emotional display on change requests, as opposed to issues, have been studied more recently. Findings exist that comments written by users and commenters submitted to issues vary in terms of sentiment, politeness and emotions (Destefanis et al., 2018), and moreover that different OSS communities may behave differently and exhibit varying levels and types of emotional display.

Having established the presence and variety of the emotional display phenomenon in OSS communities, more recent research has examined its effect on those communities. Particularly relevant is the study of positive emotional display over time, which has made several findings related to our research question (Werder, 2018):

a. That the inception of an OSS project marks a high point of positive emotional displays among the community members;
b. That team emotional display decreases over time;
c. That it is possible to create predictive models of the effect on time on team emotional display

Notably, Werder recognises the limitations of studying only positive displays of emotion, suggesting that a more wholesome predictive model may only be constructed once both positive *and* negative emotional displays have been studied.

The effects of negative emotional display on OSS communities have been partially explored by means of empirical analysis of the effect of accepted change requests. Notably, (Ortu et al., 2019) looked at how emotionally charged comments on issues affect the acceptance rate (how often a change request is accepted) of the linked change requests for that issue. Evidently, change requests resolving issues with higher level of anger, sadness and arousal are less likely to be accepted, while change requests resolving issues with higher levels of valence and joy are more likely to be accepted (Ortu et al., 2019). The two settings are distinct in that issues represent the process of reporting bugs and problems with the software or request new functionality, while contribution requests are concrete and objective changes that address issues or implement new functionality. Thus, while issues are reporting and discussion mechanisms, pull requests are evaluation mechanisms for concrete change proposals with potentially direct impact on the project and its community.

Based on that empirical observation of the communication between a change requester and a maintainer, we can formulate two initial hypotheses.

***[H1]**: Change requests with comments containing negative emotional display are less likely to be accepted.*

[H1] observes the relationship between negative emotional display and the acceptance of contribution (that is, boolean outcome of an accepted or not-accepted contribution). Acceptance of a contribution request depends on the judgement of the maintainer or maintainers, rather than business objectives, legal or other criteria. Thus, emotional bias toward the request or contributor may be a factor in the acceptance decision of the maintainer. This hypothesis may disprove the existence of such bias, should it not hold. Should a relationship between negative emotional display and change request exist, the boolean outcome of the change request can be the primary and most impactful manifestation of such relationship.

In a black box model, the input (negative emotional display) and output (change request result) can thus be used to evaluate whether or not there is a statistically observable relation between those two constructs.

***[H2]**: Change requests with comments containing negative emotional display are more likely to exhibit longer approval times.*

[H2] studies the correlation that may or may not exist between negative emotional display and contribution approval times. While H1 supposes a quantitative expression of a hypothetical relationship, H2 assumes an impact on time. The hypothesis, should it hold true, may be used for further studies of conflict resolution mechanisms in environments characterised by voluntary work contributions. In summary, it informs about a relation between emotional display and time can be observed.

If either of these hypotheses is true, it's immediate effect on the number of contributions made may impact on the continuing engagement of the contributor with the open source project (Ortu et al., 2019).

Research strategy and methodology

The research methodology follows a data-driven method for qualifying the communities that are studied. The study is designed to research whether the comment sentiments affect the likelihood of a change request receiving approval. The study is also investigating if negative sentiments expressed in a change request increase the likelihood that such requests takes more time to be approved.

I build our dataset by identifying the 50 communities with the highest number of contributors by extrapolating GitHub activity data (Ortu et al., 2019). Freeform text, referred to as comments, from an actor in the Pull Request lifecycle (About Pull Requests, n.d.) is analysed using machine learning methods from previous experiments (Werder, 2018) and (Murgia et al., 2018). Correlation coefficient indexes are calculated as statistical tests to study the display of negative emotion and its relationship with other project variables. We establish a relationship between negative mean display of emotion and merge time using Pearson's, Kendall's and Spearman's correlation coefficients. Pearson's correlation coefficient (PCC) measures linear correlation between two variables. Kendall (τ coefficient) and Spearman (ρ coefficient) correlation coefficients are both measures of rank (i.e. non-linear) correlation.

Data & empirical setting

Before describing the data source, it is important to understand the different roles that members of the OSS community play at different stages within a project, because these are the creators and participators in the emotionally-powered exchanges.. Those roles, as mentioned earlier, are distinctly a phenomenon of the voluntary open source contribution practice, rather than prescribing to the traditional form-based model of hierarchical authority.

Each person involved in an OSS contribution effort plays at least one of the following roles, which may at times overlap. We recognise the roles of contributors, change requesters, reviewers and maintainers[1]. The roles of reporter, commenter and maintainer are also recognised within the context of bug reporting, feature requests or discussions[2].

A collective term for the bug reports, feature request and discussion artifacts is referred to as an issue in this thesis.

Within the open source community stands GitHub, a platform known for its "social coding" features and as the largest code host in the OSS world (Vasilescu et al., 2014). In 2018, the platform reached 100 million *repositories*, a term describing a single project's code and artifacts within a version control (VC) environment. At that point, 31 million developers had made 1.1 billion contributions to the OSS world on the platform. Each repository is owned (controlled) by exactly one organisation (parent account).

As such, the GitHub platform represents a suitable data source for observing and analysing the interactions and emotional displays of open source contributors.

---

[1] See Appendix 3.1

[2] See Appendix 3.2

Data source

On GitHub, the request for a contribution, which then triggers the events described earlier, is called a Pull Request. Within a request for contribution, there are a number of events that determine the information structure and are used in the process of accepting or rejecting the contribution. An event has a corresponding action, which is a property of the action performed to that event [3].

For the purposes of conducting this research, the events listed below have been used:

Event	Use
PullRequestEvent	• Select the 50 most active organisations • Select the Pull Requests for a given organisation • Select the merged Pull Requests for a given organisation • Select the comments for a given Pull Request
IssuesEvent	• Select the 50 most active organisations
IssueCommentEvent	• Select the 50 most active organisations
PullRequestReviewCommentEvent	• Select the 50 most active organisations • Select the comments for a given Pull Request
PushEvent	• Select the 50 most active organisations
PullRequestReviewEvent	• Select the comments for a given Pull Request

Table 1: Mapping of GitHub events to data operations

[3] See Appendix 4.1

Using an aggregated database of the GitHub Events API, we gathered data about communication on Pull Requests for the 50 largest organisations by number of contributors in the year 2020[4].

org	authors	pr_authors	issue_authors	comment_authors	review_authors	push_authors
microsoft/	96545	11783	54750	71516	5040	3566
microsoftdocs/	45321	9904	29191	23382	905	721
google/	34645	15859	13136	20501	2433	1419
apache/	25465	10343	10554	18863	5682	1539
facebook/	24890	3008	8403	20516	849	187
azure/	24236	5038	13283	18428	2837	1643
dotnet/	22701	2981	13288	17422	1431	468
flutter/	21195	1405	9543	17953	591	156
tensorflow/	17889	2349	8576	15284	1042	287
docker/	17852	858	7912	12474	212	50
aws/	16597	2459	8047	13363	1155	585
kubernetes/	15254	3676	6556	12936	1958	58
hashicorp/	12787	2556	6318	9437	704	336
alibaba/	12386	1450	7702	8076	331	302
home-assistant/	12385	2419	5158	10710	1039	73
nextcloud/	11629	935	6098	9353	273	139
terraform-providers/	11307	2022	5066	8904	820	250
firstcontributions/	10739	10612	91	658	31	5
vuejs/	9583	1879	4060	7160	334	45
angular/	9544	902	4131	7745	341	47
react-native-community/	9499	872	3117	8023	263	87
pytorch/	9361	1831	4755	7321	812	247
elastic/	8502	1774	4318	6623	806	514
ant-design/	8497	850	4800	6141	212	61
firebase/	8396	546	3404	7191	246	150
awslabs/	8310	2092	4192	5902	795	666
learn-co-students/	8055	7945	85	121	12	247
laravel/	7764	2097	3215	5881	320	15
facebookresearch/	7629	1027	4925	5427	293	275
mozilla/	7278	1619	3727	5452	775	415
aws-amplify/	7205	474	3314	6224	146	68
rust-lang/	7131	2143	3598	5631	1081	132
valvesoftware/	6950	97	2387	6249	19	15
grafana/	6810	1061	3005	5506	389	132
golang/	6542	738	3888	4905	47	16
gatsbyjs/	6528	1740	2259	5478	517	215
zero-to-mastery/	6514	6420	247	747	46	32
github/	6480	2741	2215	3967	692	294
googlecloudplatform/	6464	2395	3129	4527	1026	697
kubernetes-sigs/	6454	1883	2975	5464	1041	76
jetbrains/	6336	3762	1606	3247	393	407
helm/	6214	1867	2336	5009	570	27
ansible/	6111	1245	3149	4423	451	85
definitelytyped/	6101	3717	924	4149	1193	32
googleapis/	5934	877	3304	4552	452	179
apollographql/	5897	829	2033	4858	204	59
vercel/	5766	930	2063	4536	284	41
homebrew/	5603	3337	1437	3486	892	46
eclipse/	5591	1463	3535	4319	804	424
ionic-team/	5577	684	2631	4285	98	44

Table 2: Level of activity, selected GitHub organisations

[4] See Appendix 2.1

Within those organisations, I identified all 649,661 Pull Requests in public repositories[5], collecting a total of 283,388 communication interactions [6]. These constitute the complete interaction on contribution requests for all public projects of the 50 largest organisations on GitHub in 2020. Additionally, I determined a status for each pull request, as either merged (true) or not merged (false), based on whether a merge event exists for a given Pull Request[7]. The 2020 dataset contains 574,616 (88.46%) merged Pull Requests and 75,045 (11.54%) not merged Pull Requests.

Data tools

The data described above is obtained from an aggregation of the GitHub Events API, sourced from the GitHub Archive in an SQL-friendly environment. We developed a number of scripts in the Python programming language that perform the following data manipulation operations: ***fetching***, ***local file storage and retrieval***, ***data cleaning***, ***sentiment analysis classification***, and ***presentation***. To do this, we use 4 script files and 4 util (assisting) classes.

Util classes

The util classes provide utility methods that are used in the main scripts, in order to reduce code complexity and adhere to the SOLID development principles [8].

[5] See Appendix 2.2
[6] SeeAppendix 2.3
[7] See Appendix 2.4
[8] See Appendix 2.5

Fetching

The fetching operation is executed first in order to collect the source data for the analysis. It is the only step in the process that interacts with the external events API discussed earlier[9].

Data cleaning

In preparation for the analysis, the "comments-cleaner.py" script performs data cleaning of the comments. The data cleaning constitutes parsing the source comment from Markdown format to plain text, stripping code tags and removing excessive white spaces. It uses the Cleaner utility class.

Sentiment analysis

After the cleaning of the data in preparation of the sentiment analysis, we identify the sentiment of the comments. We follow established methodologies for extracting sentiments from a freeform text comment (Destefanis et al., 2018), by using ordinal variable values to express sentiment across three categories. The machine learning-based sentiment extraction approach has been used in previous studies of developers' interactions on GitHub (Werder, 2018) and (Ortu et al., 2019).

To perform the analysis, the standard Python Natural Language Toolkit (NLTK) is used. In particular, the "comments-analyser.py" script, we produced a sentiment analysis of all comments using the Vader sentiment analysis utility of the NLTK.

The sentiment analyser determines the strength of three sentiments on a score of 0 to 1. The three sentiments are: ***positivity***, ***neutrality***, and ***negativity***.

The table below provides examples of comments and their sentiment strength along the three sentiments:

[9] See Appendix 2.6

Comment body	Positivity	Neutrality	Negativity
Wrong.	0.0	0.0	1.0
Worthless warnings	0.0	0.0	1.0
:(	0.0	0.0	1.0
This is not ideal, but it looks like Angular starters do this, so ¯\(ツ)/¯	0.0	0.926	0.074
Create-react-app puts the types in dependencies as well, so it was updated to better match CRA since it's used to create the app.	0.270	0.730	0.0
Very minor point but maybe above the separator on l.55 is better? It would immediately follow the instruction on zone-flags.ts that way.	0.061	0.882	0.057
wow, 👍	1.0	0.0	0.0
Yes! Thanks.	1.0	0.0	0.0

Table 3: Emotional display categorisation subset

Additionally, we categorise comments in Pull Requests based on the pull request status, either “merged” or “not merged”, for the purposes of testing H1 and H2. Merged Pull Requests represent contributions accepted into the source code of the repository, while not-merged pull requests have either a pending or a closed event associated with them.

We use three correlation tests to develop a clearer understanding of the relationship between merge time and mean negative emotional display. The correlation tests constitute the Pearson correlation coefficient (linear correlation), the Kendall rank correlation coefficient (t test) and Spearman’s rank correlation coefficient (monotonic function test).

Presentation

For analysing the results, Python's pandas data analysis library is used. It is an open source data manipulation and analysis tool, which we also use for the generation of the various figures throughout.

Analysis and results

As stated above, the total number of comments analysed is 283,338. The table below provides a summary of the entire dataset for the sentiment analysis illustrated in the previous section. It uses the three sentiments for positivity, neutrality and negativity.

Dataset summary

Level of emotional display in Pull Request comments

Measure	Negative	Neutral	Positive
Comment displays one emotion only	85 (0.03%)	97,761 (34,49%)	1234 (0.44%)
Mean	0.02868	0.75866	0.07371
Median	0.0	0.891	0.0
Mode	0.0	1.0	0.0

Table 4: Emotional display summary

A study of the table above reveals that a majority of comments (65.15%) display a mixture of at least two displays of emotion, followed by 34.49% of purely neutral comments and a total of less than 0.5% of entirely negative or entirely positive comments.

Level of emotional displays in Pull Request comments by community

The levels of negative, neutral and positive emotional display differ across communities[10]. In the 8,437 repositories of the 50 organisations that were studied, 8,060 (95.5%) were within less than half standard deviations of the mean display of negative emotions. 371 (4.5%) of repositories had half a standard deviation more negative displays, 196 (2.32%) had one or more standard deviations, and only 53 (0.63%) had displayed two or more standard deviations negative displays[11]. The close proximity of the level of negative emotional displays across repositories[12] suggests that project-specific variables, such as the organisational culture, technological stack and community size, are unlikely variables for determining the characteristics of negative emotional displays of a community.

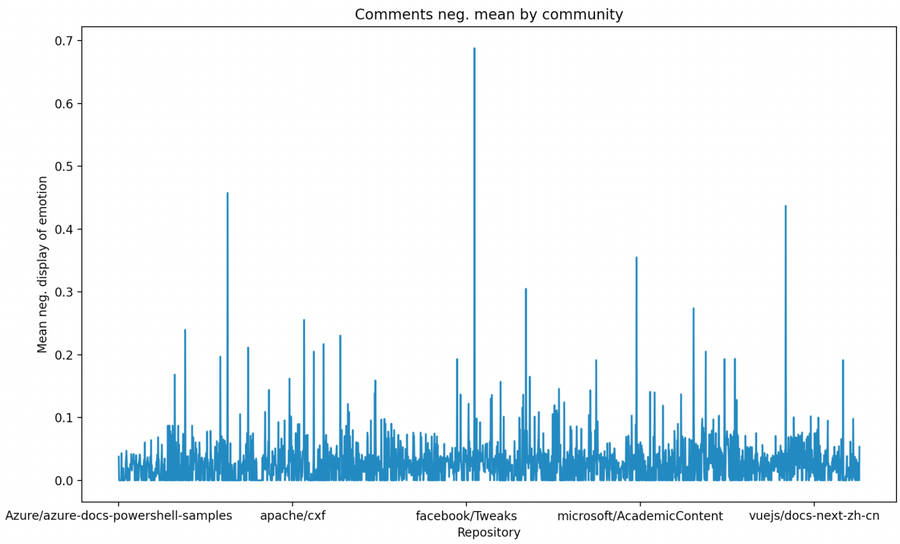

Figure 3: Comments neg. mean by community

[10] See "Comments neg. Mean by community" line and histogram figures.
[11] See Figure 4.
[12] See Figure 3.

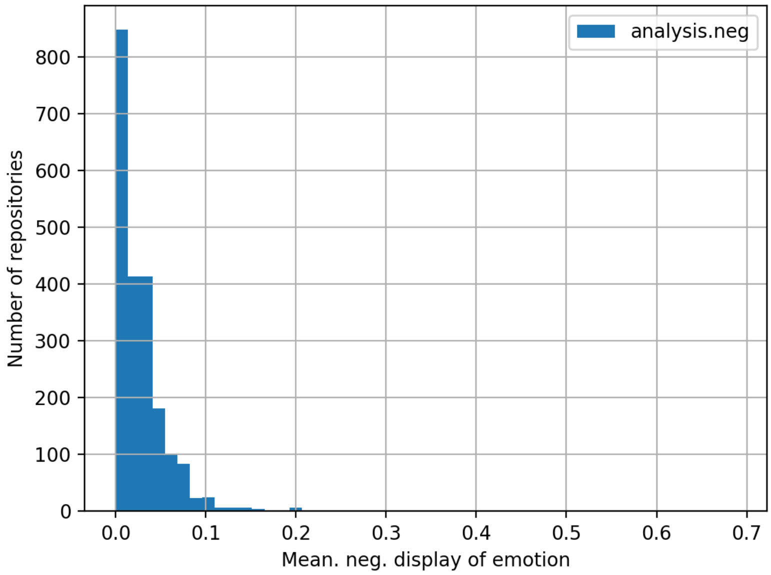

Figure 4: Comments neg. mean by community (aggregated)

Variance and std. deviation of emotional displays in Pull Request comments by community

We observe a slightly higher variance in the display of positive emotions amongst communities, with a standard deviation that is 55.8% higher for positive displays versus negative.

Measure	Negative	Neutral	Positive
Variance	0.001388	0.080955	0.080955
Std. deviation	0.037266	0.284526	0.058070

Findings

Displays of negative sentiments do not reduce the change request acceptance rate

H1: Change requests with comments containing negative emotional display are less likely to be accepted.

To investigate the effect of negative emotional display on the Pull Request status, the Pull Requests are categorised as merged or not merged. We find that the mean negative display of emotion is 9% lower among not-merged pull requests compared to merged pull requests. This could suggest that negative display of emotion in comments does not reduce the rate of accepting a contribution into a project's code. This does not necessarily suggest a causation relationship between the two. Pending or closed contribution requests may require increased development and review efforts, or be aimed at resolving more complex functionalities. These factors may in themselves impact upon the outcome of the accept/reject decision.

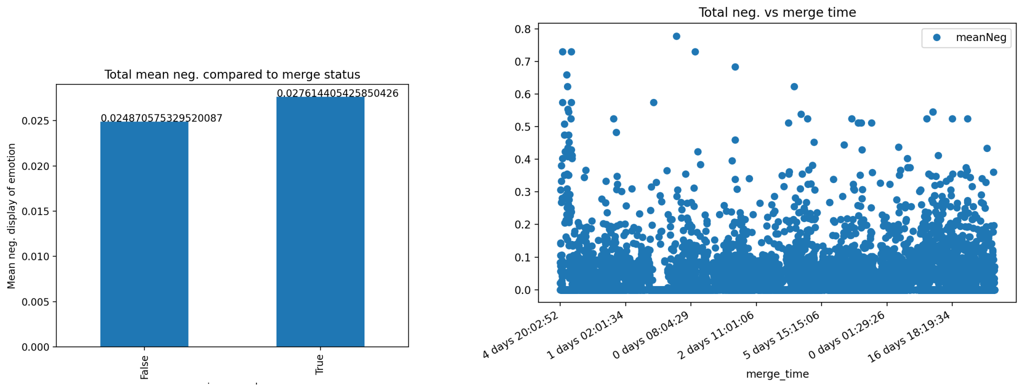

Figure 6: Mean neg. and merge status

High levels of negative emotional display increase the approval time of a change request

H2: Change requests with comments containing negative emotional display are more likely to exhibit longer approval times.

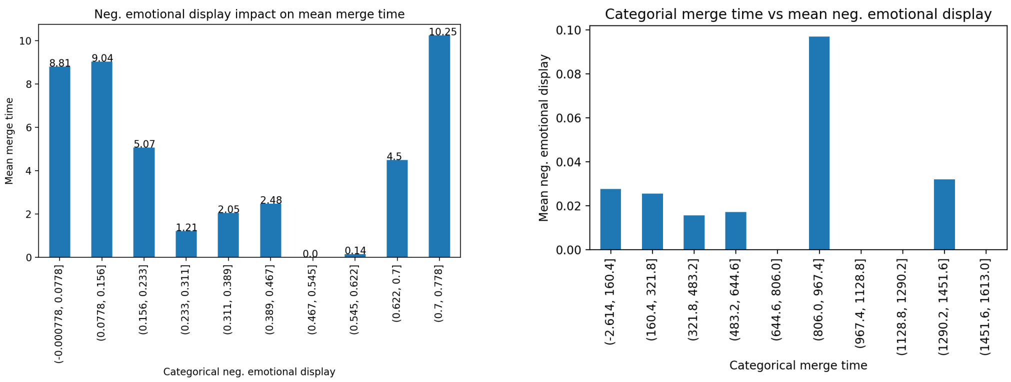

Figure 7: Negative emotional display impact on merge time

We find that the mean merge time for a Pull Request varies depending on the level of negative emotional display in the Pull Request's comments. Pull Requests with negative emotional display of .7 or higher had an average merge time of 10.15 days, which is 17.5% above the average 8.64 days.

Observations show that mean merge time of Pull Requests with negative emotional display between the mean and 1 standard deviation soars to 11.2 days, a steep increase of 29.6%.

Pearson's correlation coefficient for mean negative emotional display and merge time

Pearson's correlation coefficient (PCC) measures linear correlation between two variables. For the comments data set, we calculate PCC -0.006111. Given a significance level α=5% (0.05) and p-value=3.0993e-13, we interpret the results as insignificant. Since insignificant PCC values around 0 indicate non-linear correlations, the conclusion is that there is no linear correlation between the merge time and the mean negative emotional display of a Pull Request.

Kendall's and Spearman's correlation coefficient for mean negative emotional display and merge time

The two values based on mean negative emotional display and merge time of the comments data set are τ=0.09423 and ρ=0.112552. Both values have α>5%, p-value(τ)=1.9859e-16 and p-value(ρ)=2.4248e-16, which indicates a statistically significant rank correlation between the two variables.

Notably, correlation values lower than 0.3 are often interpreted as loose correlation.

These findings suggest that marginally more negative Pull Requests correlates to longer approval times than the more infrequent contributions that exhibit stronger displays of negative emotion. In each instance the mean approval time of Pull Requests with higher-than-average negative emotional display increases.

Directions for future research

This study had the purpose to study how, if at all, negative emotional display in communication affect the collaborative work of individuals outside the firm-based model. To do this, it studied the setting of public open source project collaboration, which includes dispersed individuals who make voluntary contributions. In this setting, I found that the strength of negative emotional display does not correlate to acceptance of contributions. In addition, the study establishes a positive correlation between the display of negative emotion and the approval outcome of voluntary contributions.

Summary of findings

Within this research, we make the following findings based on the evidence from the analysis of open source contributions:

2. The level of negative emotional display in collaborator communication does not decrease the likelihood of contribution approval [H1 does not hold]
3. Above-average negative emotional display correlates the contribution approval time [H2 does hold]

Importance of findings

My study contributes to our knowledge of emotional display in voluntary work collaboration settings. By studying contribution requests, it confirms earlier findings regarding the relationship of emotional display in issue reporting and contribution acceptance within a new empirical setting (Ortu et al., 2019). Furthermore, it closes an existing gap in the literature regarding the relationship of time and emotional display on such contributions (Werder, 2018). The new information that it adds shows the difference in observations between positive emotional displays, that are associated with short-term benefits, and display of negative emotions that I find increases the processing time of a contribution request.

Understanding both positive and negative emotional display gives a holistic picture of the nature of emotionally charged conversations. It brings the research community one step closer to developing a mixed emotional display model to understand communication in voluntary collaboration communities (Werder, 2018).

The study also confirms what other recent studies have found (e.g. (Ortu et al., 2019)) that emotionally charged communication impacts the open source software development process. In that way, it is an illustration of how to apply existing sentiment analysis capabilities to the fields of open source and emotional display research.

Limitations and future research

The findings in this research are conclusive to justify the argumentation made earlier, and to determine that H1 does not hold and H2 does hold. Nevertheless, there are three main limitations which future research may address.

Community size and behavioural bias

The dataset collected in this research only captures a small fraction of GitHub open source collaborations, which in itself is one of multiple platforms and methods for voluntary collaboration on OSS projects. As stated above, developers on the GitHub platform have made over 1.1 billion contributions, thus the limited subset studied here may not be representative of the total number of communication exchanges on the platform, or on other platforms.

When presenting the dataset, I commented that the contributions of the 50 largest organisations in 2020 were collected. Large open source communities may not necessarily communicate in the same way as medium-sized or small communities. The level of formality, procedures for reviewing and accepting contributions and relationships between the actors may have an impact on the communication style and exchanges, the processing time and the outcome state of a contribution request.

In addition, a longitudinal study may reveal another aspect of the relationship between negative emotional display and change request outcome and timing. For instance, rejected contribution requests may be re-opened with sufficient changes that do not stipulate negative displays and ultimately receive acceptance by a community's maintainer(s). This will give future research insights on the consequences of strong negative emotional display in accepted or rejected contributions. This next step will study not only the phenomenon itself, but its effects and consequences on the communities that it affects.

History of request contribution

As stated, emotional displays in issues have been studied previously (Ortu et al., 2019). The current paper contributes to our understanding of open source communication at the level of evaluating a contribution request. Future research may look at whether a trail exists over the entire reporting-contributing-evaluating cycle, analysing both issues and linked pull requests. This will give a clearer picture whether negative emotion is displayed at the time of reporting a problem, or that it only emerges in discussions on a particular solution in the form of a contribution request. Knowing this would allow for research and communities to not only understand this communication better, but also take proactive steps in using its benefits and addressing its challenges.

Reflection and critical evaluation

In conducting a reflection and critical evaluation, there are two aspects that need consideration: process and substance.

Process reflection

The process of executing this research largely consists of 5 components: identification of the research area, formulation of the research question, review of related literature, collection of empirical data and analysis of results.

Identification of the research area

The identification of the research area happened with inputs from me as the person conducting the research, my thesis supervisor, and the company at an early stage of the thesis. The numerous meetings facilitated communication between the three sides, however in retrospect bridging the gap between commercial interest and academic requirements for a thesis could have been addressed more appropriately early on should circumstances have allowed for physical meetings between the supervisor and the company. That way I could familiarise the supervisor with the day-to-day workings of the company, as well as allow the firm to better grasp the academic nature of the requirements for a thesis.

Formulation of the research question; review of related literature

The formulation of the research question, alongside the literature review, is imaginably one of the most difficult steps for students, given our little exposure to scanning, filtering and analysing academic literature outside of formal classes. I often familiarised myself with research papers that I used in order to formulate a research question within the area of study (open source collaboration), but later realised the difference between familiarisation and understanding. In the process of doing this, I learnt that knowing what the research is about is not the same as truly understanding its ins and outs and, subsequently, being able to evaluate its proximity to my own work. While in the end in my judgement the literature review provides a fine discussion of the related work, this came at the price of weeks of time.

Collection of empirical data

After identifying the research problem and conducting the literature review came the process of finding an appropriate dataset for the empirical part of the study. The open source collaboration platform GitHub was chosen based on its merits as the largest such platform and its previous use in related research. A number of platforms that provided historical data dumps were evaluated based on their technical merits and documentation, before selecting the tool that allowed the collection of the information in the right format for further processing. The process was smooth largely because of my prior technical knowledge and experience, which helped in evaluating the different platforms and coming up with a solution for extracting the necessary information.

Analysis of results

The analysis of the results provided a mixture of outperforming my expectations for my progress in some parts and facing more challenges than expected in others. The scripts' implementation in small, reusable and self-documenting parts reduced the time for the analysis, by making it easier to repeat challenging parts more often. In retrospect, it also vastly increases the reproducibility of the study for further analysis and verification of findings. The challenging component here was the qualitative analysis and interpretation of results. In particular, the transition from a merely statistical review of the results, to academic analysis and critical evaluation required a few loops of feedback and improvement between me and my supervisor.

Substance reflection

Similar to the reflection on the process, there are a number of topics to critically evaluate in terms of the results of the study: the frequency of negative emotional display; and the results of H1.

Frequency of negative emotional display

The frequency of negative emotional display was found to be substantially lower than that of both neutral and positive emotional display. The current dataset, selected on the basis of community size, included 283,338 comments, of which only 0.03% exhibited purely negative emotion and the mean display of negative emotion for all comments was 0.028, compared to 0.073 and 0.758 for positive and neutral respectively. This dataset was selected with the intention of studying those large communities, irrespective of their negative emotional display. However, a subset with bias towards negative emotions could reveal broader impacts of negative emotional display on the acceptance rate and merge time. If I were to conduct the study again, it would include both the current findings as a benchmark, as well as a set of open source communities displaying higher levels of negative emotion and then contrast and compare the different results of these exercises.

Results for H1

The results of H1 were surprising to the extent that we did not find evidence of negative emotional display affecting the acceptance rate of contributions. That is not, however, to say that a different look at this phenomenon may not reveal a correlation, or even a causation, between the two. One of the suggestions outlined already is to study whether the display of negative emotion happens outside of Pull Request comments, for example on GitHub issue reports or even on communication platforms outside of the version control hosting software altogether.

The lack of evidence also hints at the possibility that negativity alone is not a deterministic factor for contribution approval. First, negativity may not be pointed at the contribution itself. Contributions implement new functionality or resolve issues, therefore the negativity expressed in comments may be related to the underlying problem that the contribution resolves. In complex projects, such as these of the 50 largest GitHub organisations tackle, third-party dependencies updates may introduce bugs and problems that need resolving, and thus any negative emotional display may be directed at those, rather than the contribution in question.

Conclusion

This paper contributes to our understanding of communication in open source projects. By summarising the current state of research, it identified a gap in the common knowledge in regards to the display of negative emotion in voluntary collaborations. The two hypotheses represented two unknowns within that gap that were actioned upon. The data collection and analysis disproved one and confirmed the other in order to reach the findings falling within the knowledge gap that was identified. The stated limitations and suggested future research point ahead to the next steps in increasing our understanding of OSS project collaborations and communication.

Bibliography

About pull requests. (n.d.). Retrieved June 20, 2021, from https://docs.github.com/en/github/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-pull-requests

Furtado, L. B., Cartaxo, B., Treude, C., & Pinto, G. (2021). How successful are open source contributions from countries with different levels of human development? *IEEE Software*, *38*(2), 58–63.

Grewal, R., Lilien, G. L., & Mallapragada, G. (2006). Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems. In *Management Science* (Vol. 52, Issue 7, pp. 1043–1056). https://doi.org/10.1287/mnsc.1060.0550

Ortu, M., Marchesi, M., & Tonelli, R. (2019, May). Empirical analysis of affect of merged issues on GitHub. *2019 IEEE/ACM 4th International Workshop on Emotion Awareness in Software Engineering (SEmotion)*. 2019 IEEE/ACM 4th International Workshop on Emotion Awareness in Software Engineering (SEmotion), Montreal, QC, Canada. https://doi.org/10.1109/semotion.2019.00017

Roberts, J. A., Hann, I.-H., & Slaughter, S. A. (2006). Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects. In *Management Science* (Vol. 52, Issue 7, pp. 984–999). https://doi.org/10.1287/mnsc.1060.0554

Vasilescu, B., van Schuylenburg, S., Wulms, J., Serebrenik, A., & van den Brand, M. G. J. (2014). Continuous Integration in a Social-Coding World: Empirical Evidence from GitHub. In *2014 IEEE International Conference on Software Maintenance and Evolution*. https://doi.org/10.1109/icsme.2014.62

Werder, K. (2018). The evolution of emotional displays in open source software development teams. In *Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering*. https://doi.org/10.1145/3194932.3194934

Murgia, A., Ortu, M., Tourani, P., Adams, B. (2018) An exploratory qualitative and quantitative analysis of emotions in issue report comments of open source systems. In *Empirical Software Engineering* (Vol. 44, Issue 1)

Bazelli, B., Hindle, A., Stroulia, E. (2013) On the personality traits of Stack Overflow users. In *Software Maintenance* (ICSM, 29th IEEE International Conference on. IEEE, pp. 460–463)

Destefanis, G., Ortu, M., Bowes, D., Marchesi, M., Tonelli, R. (2018) On Measuring Affects of GitHub Issues' Commenters. In *IEEE* 2019

Appendices

Appendix 1: Open source code used for this thesis

The entire source code used for the data collection, data cleansing, sentiment analysis and presentation is available as an open source project in the following GitHub repository:

https://github.com/I-Valchev/leiden-thesis-github

Appendix 2: Code samples

2.1 Identifying the 50 largest organisations by number of contributors in 2020

```
SELECT
    lower(substring(repo_name, 1, position(repo_name, '/'))) AS org,
    uniq(actor_login) AS authors,
    uniqIf(actor_login, event_type = 'PullRequestEvent') AS pr_authors,
    uniqIf(actor_login, event_type = 'IssuesEvent') AS issue_authors,
    uniqIf(actor_login, event_type = 'IssueCommentEvent') AS comment_authors,
    uniqIf(actor_login, event_type = 'PullRequestReviewCommentEvent') AS review_authors,
    uniqIf(actor_login, event_type = 'PushEvent') AS push_authors
FROM github_events
WHERE event_type IN ('PullRequestEvent', 'IssuesEvent', 'IssueCommentEvent', 'PullRequestReviewCommentEvent', 'PushEvent')
AND (created_at BETWEEN '2020/01/01' AND '2020/12/31')
GROUP BY org
ORDER BY authors DESC
LIMIT 50
```

2.2 Identifying the Pull Requests of the selected organisations

```
def getPRsForOrg(org):
  return [["repo_name","number"], """
    SELECT repo_name, number
    FROM github_events
    WHERE LOWER(SUBSTRING(repo_name, 1, POSITION(repo_name, '/'))) IN ('%s')
      AND event_type = 'PullRequestEvent'
      AND action = 'opened';
    """ % org]
```

2.3 Identifying the comments of the selected Pull Requests

```
def getCommentsForRepo(repo, prs):
  return [["repo_name", "number", "body"], """
    SELECT repo_name, number, body
    FROM github_events
    WHERE repo_name = '%s'
          AND event_type IN ('PullRequestEvent', 'PullRequestReviewCommentEvent',
'PullRequestReviewEvent')
      AND action in ('opened', 'created')
      AND review_state = 'none'
      AND number in (%s);
  """ % (repo, ','.join(prs))]
```

2.4 Identifying the merge status of a Pull Request

```
def getMergedPRsFrOrg(org):
       return [["repo_name", "number", "merged_at"], """
       SELECT repo_name, number, merged_at
       FROM github_events
       WHERE LOWER(SUBSTRING(repo_name, 1, POSITION(repo_name, '/'))) IN ('%s')
       AND event_type = 'PullRequestEvent'
       AND merged_at > '2020-01-01 00:00:00' and merged_at < '2021-01-01 00:00:00';
       """ % org]
```

2.5 Utility classes, operations and properties of the extraction and analysis tool

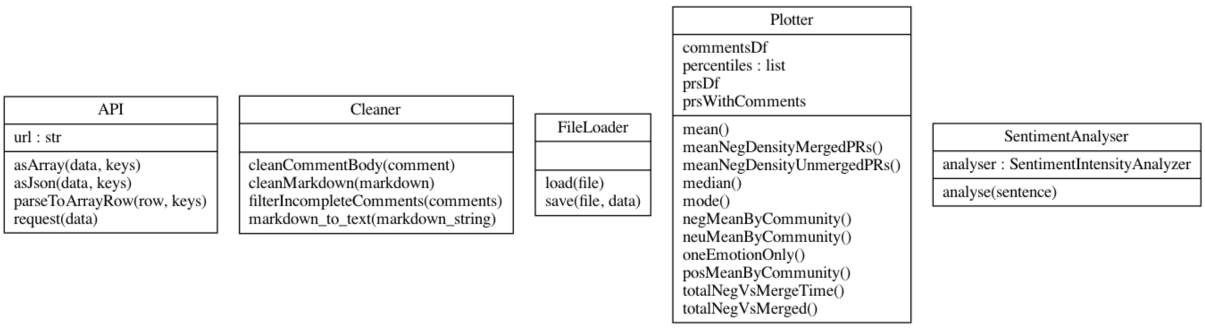

Figure 2: Utility classes, operations and properties

2.6 Fetching operation steps

The fetching operation collects the necessary data in four steps, in which order is significant:

Step	Script file	Responsibility
Identification of organisations	orgs-community-size.py	Outputs a json file containing the 50 largest orgs on GitHub in 2020
Identification of Pull Requests	orgs-pull-requests.py	Outputs a json file containing all the Pull Requests of the orgs sourced in the previous step.
Identification of comments	comments.py	Outputs a json file containing all comments associated with Pull Requests sourced in the previous step
Identification of merged Pull Requests	orgs-pull-requests-merged.py	Outputs a json file containing all merged pull requests sourced from the Identification of Pull Requests step.

Appendix 3: OSS contribution roles

3.1 Contribution roles

- Contributor: a person who makes a material change to the OSS project
- Change Requester: a person who requests the contributor’s changes to be incorporated into the OSS project
- Reviewer: a person who performs a qualitative evaluation of the requested changes

- Maintainer: a person who takes the final decision of either accepting or rejecting the change request

3.2 Issue-related roles

- Reporter: a person who reports/opens the bug, feature request or discussion
- Commenter: a person who contributes to the bug report, feature request or discussion in comment format
- Maintainer: a person who takes the final decision of either acknowledging the bug, feature request or discussion, or closing it.

Appendix 4: GitHub API

4.1 API Events and actions

event_type	action
CommitCommentEvent	none
IssuesEvent	opened
IssuesEvent	closed
IssuesEvent	reopened
CreateEvent	none
PullRequestEvent	opened
PullRequestEvent	closed
PullRequestEvent	reopened
MemberEvent	added
PushEvent	none
DeleteEvent	none
ForkEvent	none
PullRequestReviewCommentEvent	created
PullRequestReviewCommentEvent	none
IssueCommentEvent	created
PullRequestEvent	synchronize
IssueCommentEvent	none
WatchEvent	started
GollumEvent	none
PublicEvent	none
PullRequestEvent	merged
PullRequestEvent	labeled
GollumEvent	created
GollumEvent	edited
IssuesEvent	labeled
ReleaseEvent	published
PullRequestReviewEvent	created
GistEvent	fork
GistEvent	update
GistEvent	create
FollowEvent	none
DownloadEvent	none
ForkApplyEvent	none
Event	none
Event	started
Event	created
Event	closed
Event	opened
Event	added
Event	edited
TeamAddEvent	none